Pr. Mohamed RHAZZAF

ECOLE MOHAMMADIA D'INGÉNIEURS

Hands-On Exercises 4

Objective:

In this exercise, you will implement a polynomial regression model and a cost function to fit a dataset. You will also visualize the results to understand how the model performs.

Create a Jupyter Notebook:

- Create a Jupyter Notebook (or a collaborative file in Google Drive).
- Name the file as student1name_student2name_TD4.ipynb.

Import Necessary Packages:

- Import the required libraries: numpy and matplotlib.

Dataset:

We will use a synthetic dataset; try to create a 10000 sample of values between -3 and 3.

Our function to predict will be $g(x) = -1.2 + 0.5x - 1.7x^2$

Try to generate the images of the given samples and add a normal distribution noise.

Define the set of functions to fit the data:

We will define the set of function to fit with our data by:

$$F = \{f_{\theta_0,\theta_1,\theta_2}(x) = \theta_0 + \theta_1 x + \theta_2 x^2 \quad \theta_0, \theta_1, \theta_2 \in R \ \}$$

- Create a hypothesis python function that return the images of a numpy data structure using as arguments the values of theta the inputs numpy structure.
- Plot in the same figure the functions g and $f_{1,2,3}$

The optimization process:

- Create the cost function related the our problem
- Get the cost of $f_{1,2,3}$, $f_{-1,2,-3}$, and $f_{1,-2,3}$
- Calculate the gradient of the cost function
- Implement the gradient descent algorithm to get the best params related to the problem.
- Plot the result function beside the dataset

More complicated situation

Remake the same thing but for the set of function:

$$F = \{f_{\theta_0,\theta_1,\theta_2,\theta_3,\theta_4}(x) = \theta_0 + \theta_1 x^{\theta_2} + \theta_3 x^{\theta_4} \quad \theta_0, \theta_1, \theta_2, \theta_3, \theta_4 \in R \ \}$$